HOW VIRTUAL BECOMES REAL

Perform a Finite Element structural analysis with Karamba (G_01.1)

Creative team
Alberto Pugnale and Stanislav Roudavski

Content developed by
Alberto Pugnale, Alessandro Liuti and Louis Gadd

Supported by the faculty
"Strategic Initiative Funding"

THE UNIVERSITY OF MELBOURNE

msd
Melbourne School of Design

FACULTY OF ARCHITECTURE, BUILDING AND PLANNING

www.msd.unimelb.edu.au

0:00 SLIDE

- In this tutorial you are going to perform with Karamba a Finite Element analysis on the Diagrid generated in tutorial G01

0:09 1 slide Karamba pre-processor, processor, post-processor

- A Finite Element analysis is performed in three subsequent steps
- First, the "pre-processor", which translates your geometrical model into a structural model; in this phase points become nodes, lines become beams and mesh become shells
    - In the pre-processor phase you also specify supports, constraints, applied loads, cross-sections and material properties
    - All of these features are called "boundary conditions"
- Second, the "processor", which performs the actual Finite Element analysis
- Third, the "post-processor", in which you interpret the analysis results, such as nodal displacements, strains or utilisation of elements
    - This information can be represented both graphically and numerically
- Now, let's start!

0:52 GRASSHOPPER

- You can use the LineToBeam component to translate the exploded segments of your Diagrid into beam elements
- Let's use a first LineToBeam component for the main gridshell beams
    - As you can see, you can feed multiple lists to LineToBeam
    - It is also relevant to label the beam set with a meaningful ID, for example "diagrid"
- Now, use a second LineToBeam component for the bracing elements, and remember to assign a meaningful ID, such as "bracing"
- You need to repeat this procedure again for the edge beams, so that you can later assign specific cross-sections and material properties to each set individually
- Note that LineToBeam requires a list of lines, and not curves, as input
- Remember also that Finite Element solvers don't automatically recognise Line-to-Line intersections as nodes. For this reason, the previous tutorial G01 explained how to generate a Diagrid made of segments that meet at the endpoints

- Now you have to use the Karamba Cross-Section component to assign different profile shapes to the three beam sets:
    - Feed the first beam set ID, "diagrid", into Cross-Section
    - Now choose a "Hollow box" profile for the main Diagrid beams
    - Bear in mind that height, width and profile thickness are expressed in centimetres

- Now use Number Sliders to pre-dimension the profile – let's say with a height of 10cm, a width of 6cm and a thickness of 0.6cm:
    - Generally this choice is based on the overall dimensions of the gridshell and personal experience. Using Sliders, however, will allow you to increase or decrease the section size easily after the analysis – so that you can design a more efficient structure

- Now Copy/Paste the Cross-Section component and Sliders to pre-dimension the bracing elements:
    - You can preserve the profile shape and reduce the height to 6cm, and the width to 4cm
- Then, Copy/Paste the Cross-Section component again for the edge beams:
    - In this case the profile dimensions have to be more significant. For instance, 20cm by 10cm

- You can now use the MaterialSelection component to assign different materials to the three beam sets:
    - By default, a standard Karamba database is provided; simply select a family of materials from the upper dropdown menu - for example steel - and a specific material from the bottom dropdown menu - such as S235.
    - The label S235 represents a yield stress capacity of 235N/mm2 or 23.5KN/cm2.
    - Connecting a panel to the component output will display the material properties, such as Young modulus and density
    - If you need to use a custom material, use the MaterialProperties component to input your own material properties

- Now let's feed the three beam IDs into MaterialSelection to make sure that your gridshell is entirely a steel structure

- Now, you need to use the Support component to define the constraints of your model and make sure the structure is statically determinate. In this tutorial, let's assume the gridshell is constrained at the boundary, that means it's supported at the nodes of the edge beams
    - It's a good habit to double-check the presence of duplicate points and remove them with the RemoveDuplicates component before feeding the point list into Support
    - The Support component allows you to constrain the 6 degrees of freedom of a structure at a given node
    - In practice, ticking the Tx, Ty and Tz buttons will suppress the translation in the x, y and z directions
    - Bear in mind that Support refers to the global coordinate system
    - In the same way, ticking the Rx, Ry and Rz buttons will suppress the rotation about the x, y and z axes
- In reality, you are likely to use only pinned joints, where rotations are allowed, and fixed joints, where both translations and rotations are constrained. This depends on your structural design, but also on the construction technology

- The last thing to define are loads
    - You can apply a gravity load with the Loads component. As you can see from the input, by default there is a value of a negative UnitZ vector, which represents gravity on earth
    - Remember that the gravity load allows Karamba to include the self-weight of the modelled elements in the calculation
    - To apply dead loads, such as cladding, and live loads you need to use a second Loads component
    - Select UniformLine from the dropdown list to apply a distributed load on the Diagrid beams
    - Use a Slider and a UnitZ vector to define load direction and magnitude
    - The magnitude should be calculated carefully considering the real dead-load and live-load values. Generally, you will find that these values fall in the range of -2 to -4 kN/m^2
    - You need to convert this load value into an equivalent linear distributed one. For the purpose of this exercise, you can apply -2kN/m
    - Now tick the "Global" button to make sure the load is oriented according to the global coordinate system
    - And connect Loads to the "diagrid" panel to apply the load
    - In this exercise, do not consider wind pressure or snow load

6:35

- Now you have to use the **Assemble** component to generate the actual structural model
  - First, you need to feed all the **LineToBeam** components (diagrid, bracing and edge beams)
  - Then, you feed **Cross-Sections**
  - Then you have **Materials, Supports** and **Loads**
  - Remember to flatten all the inputs

- Now you can temporarily use **ModelView** and **BeamView** to visualise the structural model in the Rhino interface
- This allows you to double-check your parameters and values before running an analysis. For instance, you can verify the position of your supports and the applied loading. You can also visualise the node numbers and beam IDs
- When you are ready, then feed the Assemble component into **Analysis.** This will perform the actual Finite Element Analysis

- Now, feed Analysis into **ModelView** and **BeamView** in order to see the analysis results
- If you click on **Displacement**, you see that the results are graphically displayed in Karamba using a colour spectrum from Yellow to Purple to Green by default. It's advisable to use a more standard palette to render displacements and utilisation of elements, with colours that range from Red to Blue
- To do so, you have to use the **ColourRGB** component, to which you feed three lists of values, respectively for R, G and B

8:34 SLIDE

- In this slide you can see a standard colour palette for a Finite Element Analysis
- Use a **Panel** and feed the following RGB values into **ColourRGB**
- Alternatively, create a palette using different **ColourSwatch** components

8:50 GRASSHOPPER

- Now that you have created a standard colour palette, feed the **ColourRGB** component into **ModelView**, and **Reverse** the list
- Make sure that the low stressed parts of the structure are rendered in blue, whereas highly stressed parts are shown red

- Now, you are ready to use the **ModelView** and **BeamView** components
  - The first thing you can do is to magnify the structure deformations in real time in the Rhino interface. Make sure that the **Deformation** checkbox is ticked and move the **Slider** to select an appropriate scale
  - You can currently visualising the structure displacements in **BeamView**. In the **ModelView RenderSettings**, you need to set the **Result thresholds** as **Absolute value** and modify the upper and lower thresholds on the basis of the maximum displacement
  - You can roll the mouse cursor over the analysis output labelled "Displacement" to read the maximum value for your gridshell
  - Here you can read about 0.5cm
  - You can set that value as the upper threshold and keep the lower threshold to 0cm or 0.2cm
  - You also need to connect a **Legend** to **BeamView** in order to read the relationship between values and colours

  - Generally speaking, a good structure will have a maximum displacement that stands in the order of a few centimetres. So, if you set the Upper threshold between 3 and 5cm, you are going to visualise the problematic parts of your structure at a glance
  - In this case, the gridshell is clearly over-dimensioned, and you can slightly reduce the cross-section of the diagrid and bracing beams. For instance, to 5cmx3cm for the diagrid, and a thickness of 0.3cm, and to 3cmx2cm for the bracing, and a thickness of 0.2cm
  - Another way to double-check the structural behaviour of your gridshell is by ticking the Utilization button in BeamView. This will render "the ratio between the normal stress at a point and the yield stress of the corresponding material"
  - Every time you switch between Displacement and Utilization, you will need to tune the thresholds to get a satisfactory render of the values. You also have to disable to ColourRGB component in order to go back to the standard Utilisation-of-element colour palette. In this case, there could also be negative values, which represent compression
  - Bear in mind that shear and buckling are not considered by Utilization in BeamView

- ModelView and BeamView allow you to visualise and export many other results. For instance, you can display axial forces, shear and bending moments by using the first submenu Section Forces in BeamView
- Now you are visualising the bending moments

11:39 SLIDE
- This is the end of tutorial G01.1. In the support master file, you will find the refined GH definition, which is linked to the other definitions of tutorials G01 and G01.2:
  - To activate or deactivate this modulus in the support master file, use the switch on the top left (ValueList)
  - The components highlighted in green are placeholders of the Diagrid input geometry, parametrised in tutorial G01
  - The central part of the definition is the Karamba pre-processor
  - The structural model parameters and variables are collected into the pink groups on the left
  - Structural supports are defined at the model boundary by default. However, you can use this part of the pre-processor to create custom openings
  - This component performs the Finite Element Analysis and is activated or deactivated by the switch on the top left
  - On the top right, you will find the post-processor, to read and interpret results
  - In the support master file there is also a cluster that plots the position and value of the nodal maximum displacement

12:38 END OF TUTORIAL G01.1